

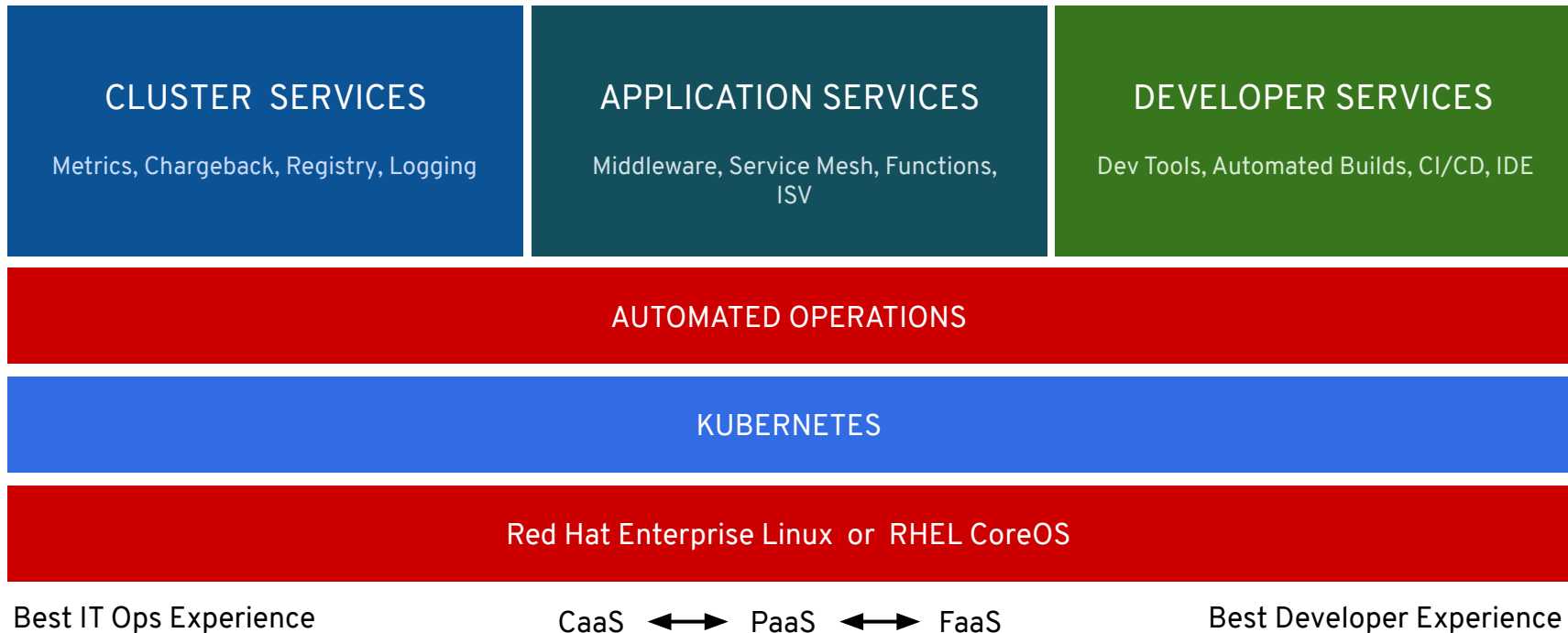


OpenShift 4 Roadmap Update

Chris Blum
Senior Architect

William Markito Oliveira
Product Management

OpenShift 4 Platform



OpenShift 4.2



Disconnected, air-gapped
installation

Full-stack Automation for
Azure, GCP, OpenStack

Migration tooling



Operators for Red Hat Integration,
Business Automation, Runtimes

Special resources operator
For GPU



Developer console with
application topology

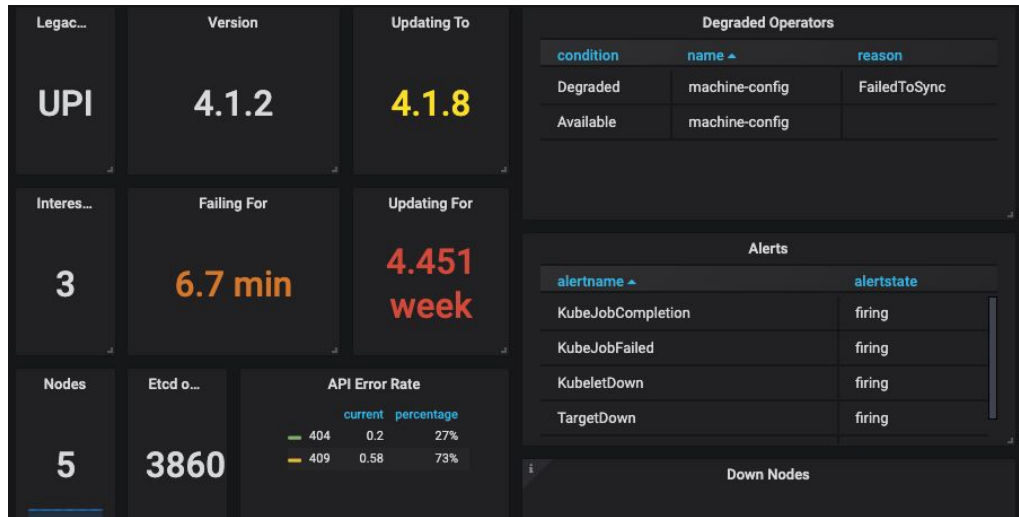
CodeReady Containers

Service Mesh fully supported

Connected Customer

- 4 months since GA
- 11 z-stream releases
- 186 bugs fixed
- 207 CVEs across RHEL+OCP

20% of bugs fixed to date have been from information obtained through telemetry



4.2 Supported Providers

Full Stack Automation (IPI)



Pre-existing Infrastructure (UPI)



Bare Metal



** Support for full stack automated installs to pre-existing VPC & subnets and deploying as private/internal clusters is planned for 4.3.*

Generally Available

Installation Experiences

OPENSIFT CONTAINER PLATFORM

Full Stack Automated

Simplified opinionated “Best Practices” for cluster provisioning

Fully automated installation and updates including host container OS.



Pre-existing Infrastructure

Customer managed resources & infrastructure provisioning

Plug into existing DNS and security boundaries



HOSTED OPENSIFT

Azure Red Hat OpenShift

Deploy directly from the Azure console. Jointly managed by Red Hat and Microsoft Azure engineers.

OpenShift Dedicated

Get a powerful cluster, fully Managed by Red Hat engineers and support.

Generally Available



Comparison between deployments methods

	Full Stack Automation	Pre-existing Infrastructure
Build Network	Installer	User
Setup Load Balancers	Installer	User
Configure DNS	Installer	User
Hardware/VM Provisioning	Installer	User
OS Installation	Installer	User
Generate Ignition Configs	Installer	Installer
OS Support	Installer: RHEL CoreOS	User: RHEL CoreOS + RHEL 7
Node Provisioning / Autoscaling	Yes	Only for providers with OpenShift Machine API support
Customization & Provider Support	Best Practices: AWS	Yes: AWS, Bare Metal, VMware, GCP(4.2), Azure(4.2)

OPENSIFT PLATFORM

Full stack automated deployments of AWS, Azure, GCP & OSP!



```

$ ./openshift-install --dir ./demo create cluster
? SSH Public Key /Users/demo/.ssh/id_rsa.pub
? Platform azure
? azure subscription id xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure tenant id xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure service principal client id xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure service principal secret *****
INFO Saving user credentials to "/Users/demo/.azure/osServicePrincipal.json"
? Region centralus
? Base Domain example.com
? Cluster Name demo
? Pull Secret [? for help] *****
INFO Creating infrastructure resources...
INFO Waiting up to 30m0s for the Kubernetes API at https://api.demo.example.com:6443...
INFO API v1.14.0+4788f50 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO Destroying the bootstrap resources...
INFO Waiting up to 30m0s for the cluster at https://api.demo.example.com:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/Users/demo/openshift-install/demo/auth/kubeconfig'
INFO Access the OpenShift web-console here:
https://console-openshift-console.apps.demo.example.com
INFO Login to the console with user: kubeadmin, password: <password>

```

Simplified Cluster Creation

Easily provision a “best practices” OpenShift cluster on Microsoft Azure

- CLI-based installer with interactive guided workflow
- Installer takes care of provisioning the underlying Infrastructure significantly reducing deployment complexity



```

$ ./openshift-install --dir ./demo create cluster
? SSH Public Key /Users/demo/.ssh/id_rsa.pub
? Platform gcp
? Service Account (absolute path to file or JSON content)
/Users/demo/.secrets/ServiceAccount.json
INFO Saving the credentials to "/Users/demo/.gcp/osServiceAccount.json"
? Project ID openshift-gce-devel
? Region centralus
? Base Domain example.com
? Cluster Name demo
? Pull Secret [? for help] *****
INFO Creating infrastructure resources...
INFO Waiting up to 30m0s for the Kubernetes API at https://api.demo.example.com:6443...
INFO API v1.14.0+4788f50 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO Destroying the bootstrap resources...
INFO Waiting up to 30m0s for the cluster at https://api.demo.example.com:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/Users/demo/openshift-install/demo/auth/kubeconfig'
INFO Access the OpenShift web-console here:
https://console-openshift-console.apps.demo.example.com
INFO Login to the console with user: kubeadmin, password: <password>

```

Faster Install

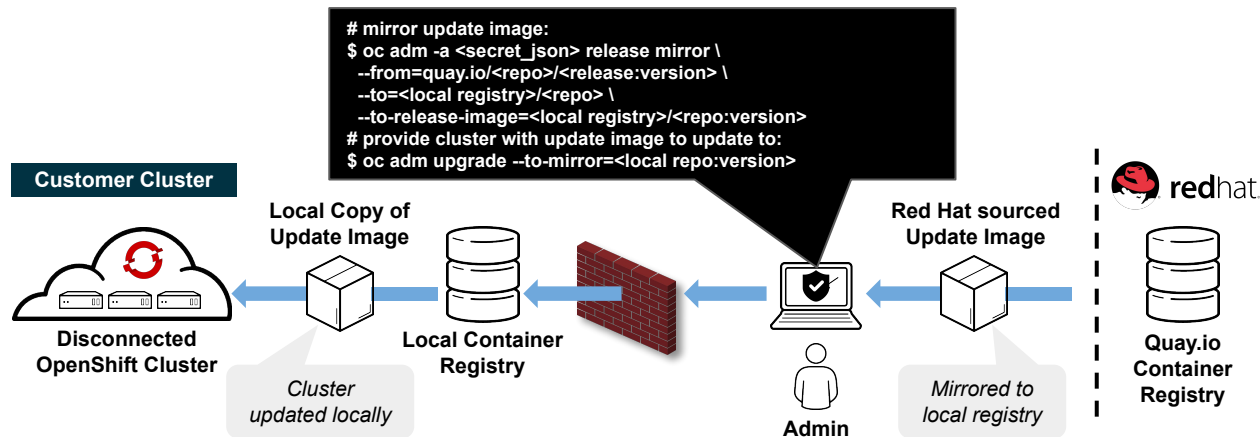
The installer typically finishes within 30 minutes

- Only minimal user input needed with all non-essential install config options now handled by component operator CRD's
- Leverages RHEL CoreOS for all node types enabling full stack automation of installation and updates of both platform and host OS content

Generally Available



Disconnected “Air-gapped” Installation & Upgrading



Overview

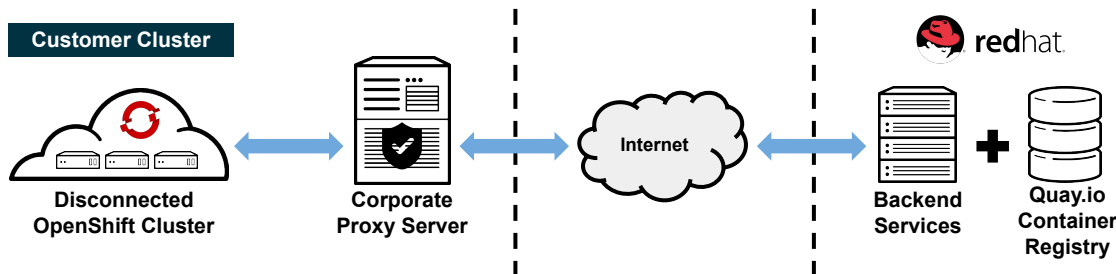
- 4.2 introduces support for installing and updating OpenShift clusters in disconnected environments
- Requires local Docker 2.2 spec compliant container registry to host OpenShift content
- Designed to work with the user provisioned infrastructure deployment method
 - *Note: Will not work with Installer provisioned infrastructure deployments*

Installation Procedure

- Mirror OpenShift content to local container registry in the disconnected environment
- Generate install-config.yaml: `./openshift-install create install-config --dir <dir>`
 - Edit and add pull secret (PullSecret), CA certificate (AdditionalTrustBundle), and image content sources (ImageContentSources) to install-config.yaml
- Set the `OPENSHIFT_INSTALL_RELEASE_IMAGE_OVERRIDE` environment variable during the creation of the ignition configs
- Generate the ignition configuration: `./openshift-install create ignition-configs --dir <dir>`
- Use the resulting ignition files to bootstrap the cluster deployment

Generally Available

Cluster-wide Egress Proxy



Overview

- 4.2 introduces support for installing and updating OpenShift clusters through a corporate proxy server
- Leverages new proxy controller within the cluster-network-operator, which is responsible for:
 - Reconciling a proxy object and writing spec > status upon successful validation.
 - Reconciling user-provided trust bundles referenced by trustedCA, validating the trust bundle certificates, merging the certificates with the system trust bundle and publishing the merged bundle to the openshift-config-managed/trusted-ca-bundle configmap.

Installation Procedure

- Installer will use PROXY* environment variables from the shell it's invoked from
- Generate install-config.yaml: `./openshift-install create install-config --dir <dir>`
 - Edit proxy information (httpProxy, httpsProxy, & noProxy) and CA certificate (AdditionalTrustBundle) to install-config.yaml
- Installer validates the provided install-config.yaml parameters, renders the necessary assets to create the cluster, and initiates the installation process based on the install method used:

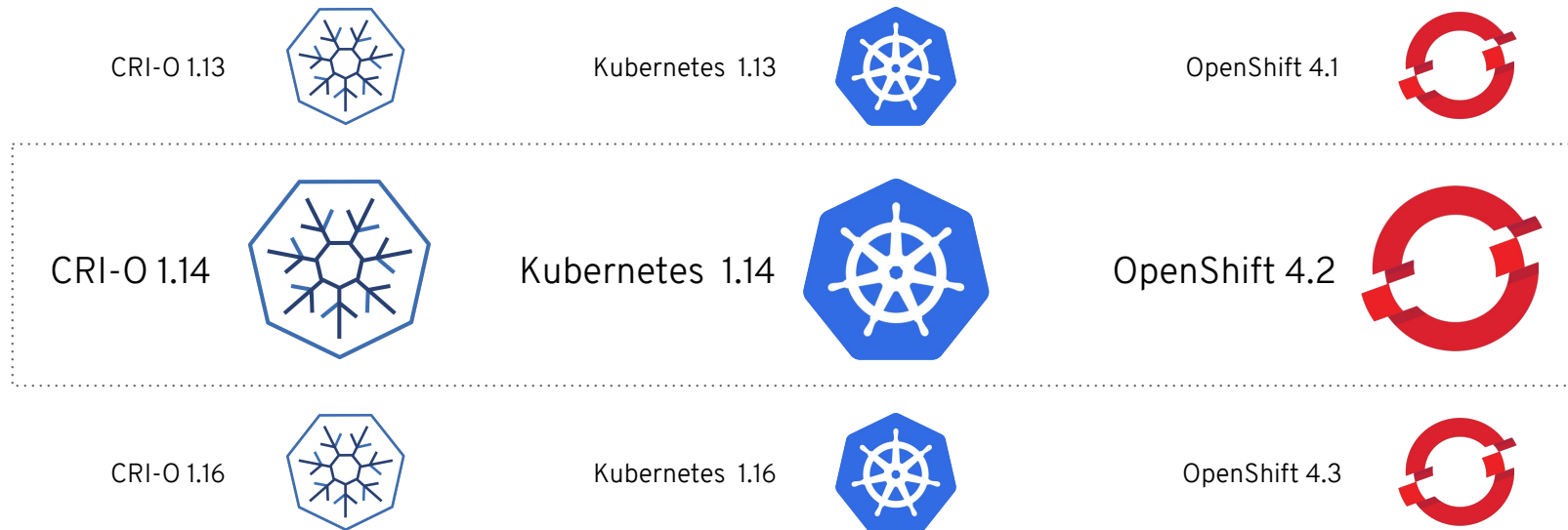
`./openshift-install create cluster --dir <dir>`

An admin with privileges can interact with the proxy object using 'oc' commands (use the 'oc edit' command to modify the proxy information.) Here is an example proxy config:

```
$ oc get proxy/cluster -o yaml
apiVersion: config.openshift.io/v1
kind: Proxy
metadata:
  creationTimestamp: "2019-08-21T22:36:49Z"
  generation: 2
  name: cluster
  resourceVersion: "24913"
  selfLink: /apis/config.openshift.io/v1/proxies/cluster
  uid: 2a344b01-d267-11f9-a4f3-025de4b59c38
spec:
  httpProxy: http://<username>:<pswd>@<ip>:<port>
  httpsProxy: https://<username>:<pswd>@<ip>:<port>
  noProxy: example.com
  readinessEndpoints:
  - http://www.google.com
  - https://www.google.com
  trustedCA:
    name: user-ca-bundle
status:
  httpProxy: http://<username>:<pswd>@<ip>:<port>
  httpsProxy: https://<username>:<pswd>@<ip>:<port>
  noProxy:
  10.0.0.0/16,10.128.0.0/14,127.0.0.1,169.254.169.254,172.30.0.0/16,api-int.demo.example.com,api.demo.example.openshift.com,etcd-0.demo.example.com,etcd-1.demo.example.com,etcd-2.demo.example.com,example.com,localhost
```

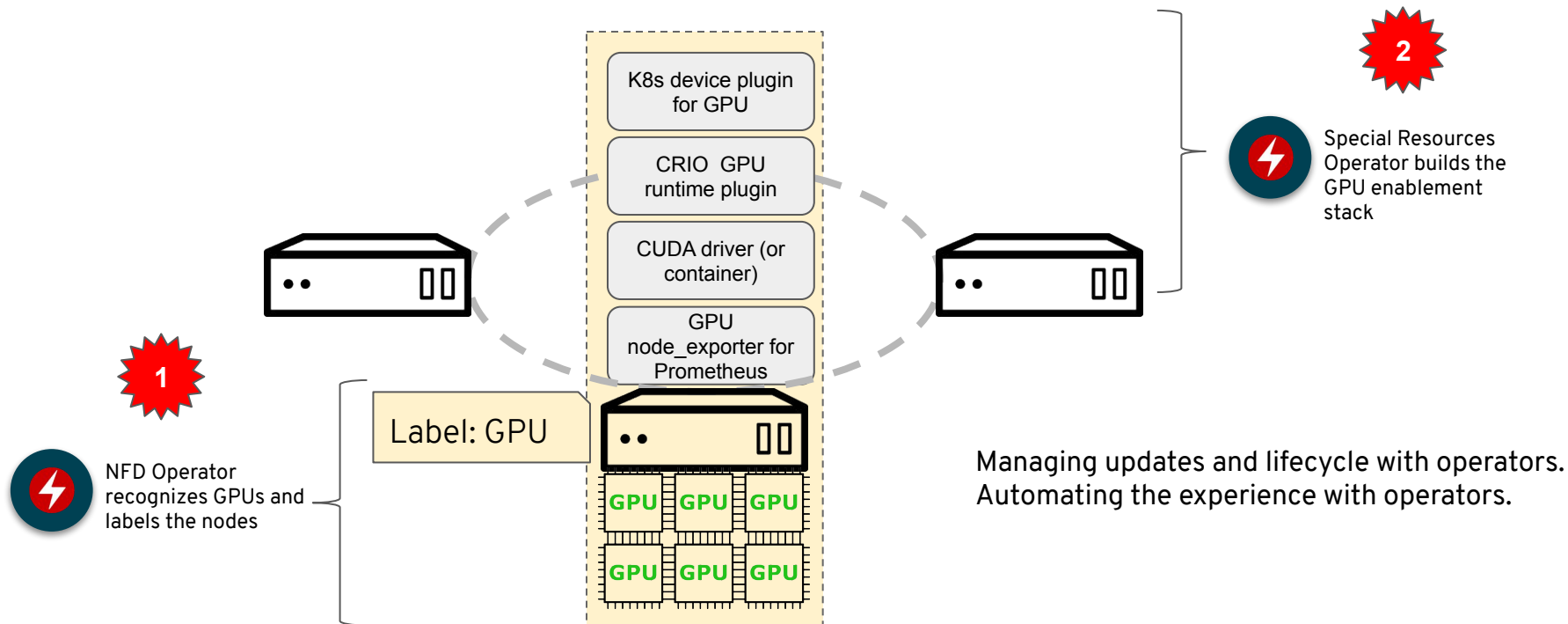
CRI-O Support in OpenShift

CRI-O tracks and versions identical to Kubernetes, simplifying support permutations



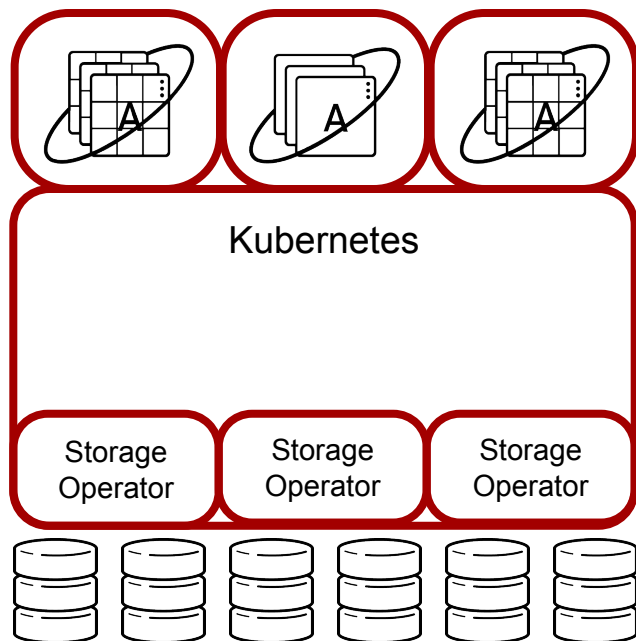
Generally Available

Enablement of GPUs in an OpenShift Cluster

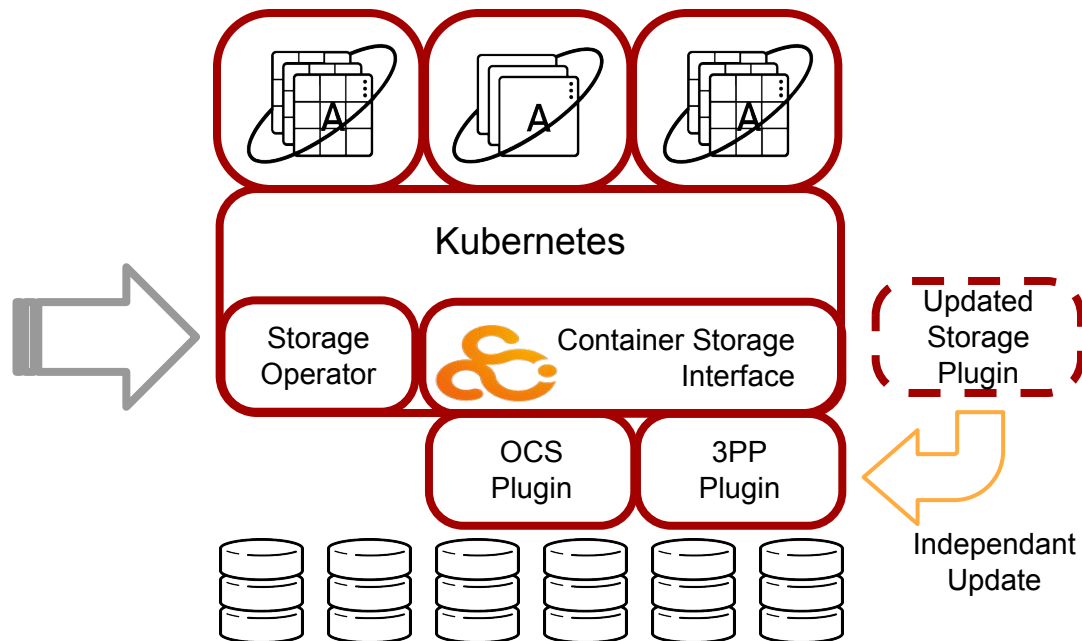


OPENSIFT CSI

4.1



4.2



CSI: ENABLING OCS AND PARTNERS

STORAGE DEVICES

Storage Focus

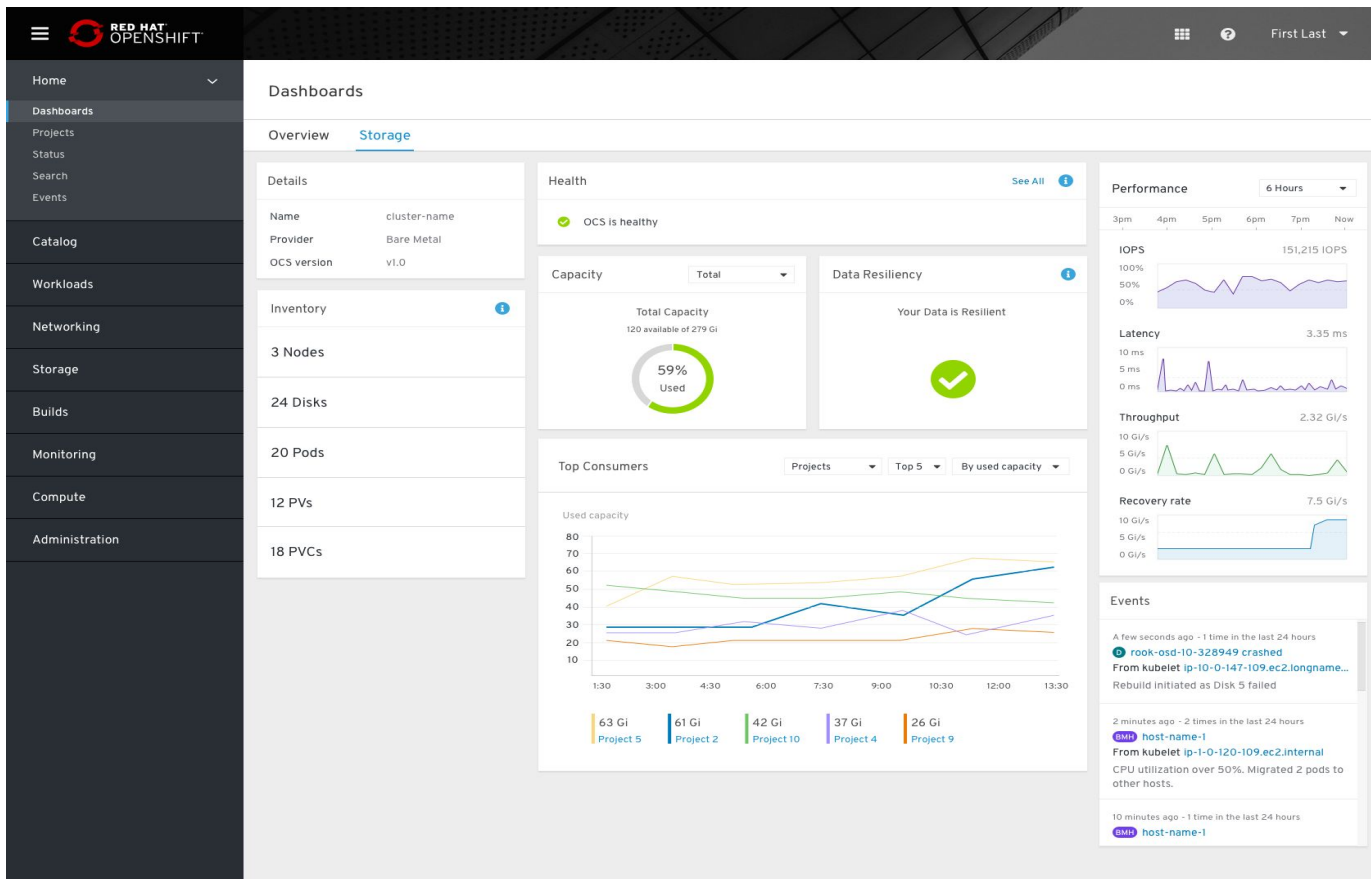
- Cluster Storage Operator
 - Sets up the default storage class
 - Looks through cloud provider and sets up the correct storage class
- Drivers themselves remain in-tree for now, CSI versions to follow later
- New GA storage in 4.2
 - Local Volume
 - Raw Block
 - Cloud providers (AWS, GCP, Azure, vSphere)
 - Local Volume

Supported	
AWS EBS	iSCSI
Azure File & Disk	Fibre Channel
GCE PD	HostPath
VMware vSphere Disk	Local Volume NEW
NFS	Raw Block NEW

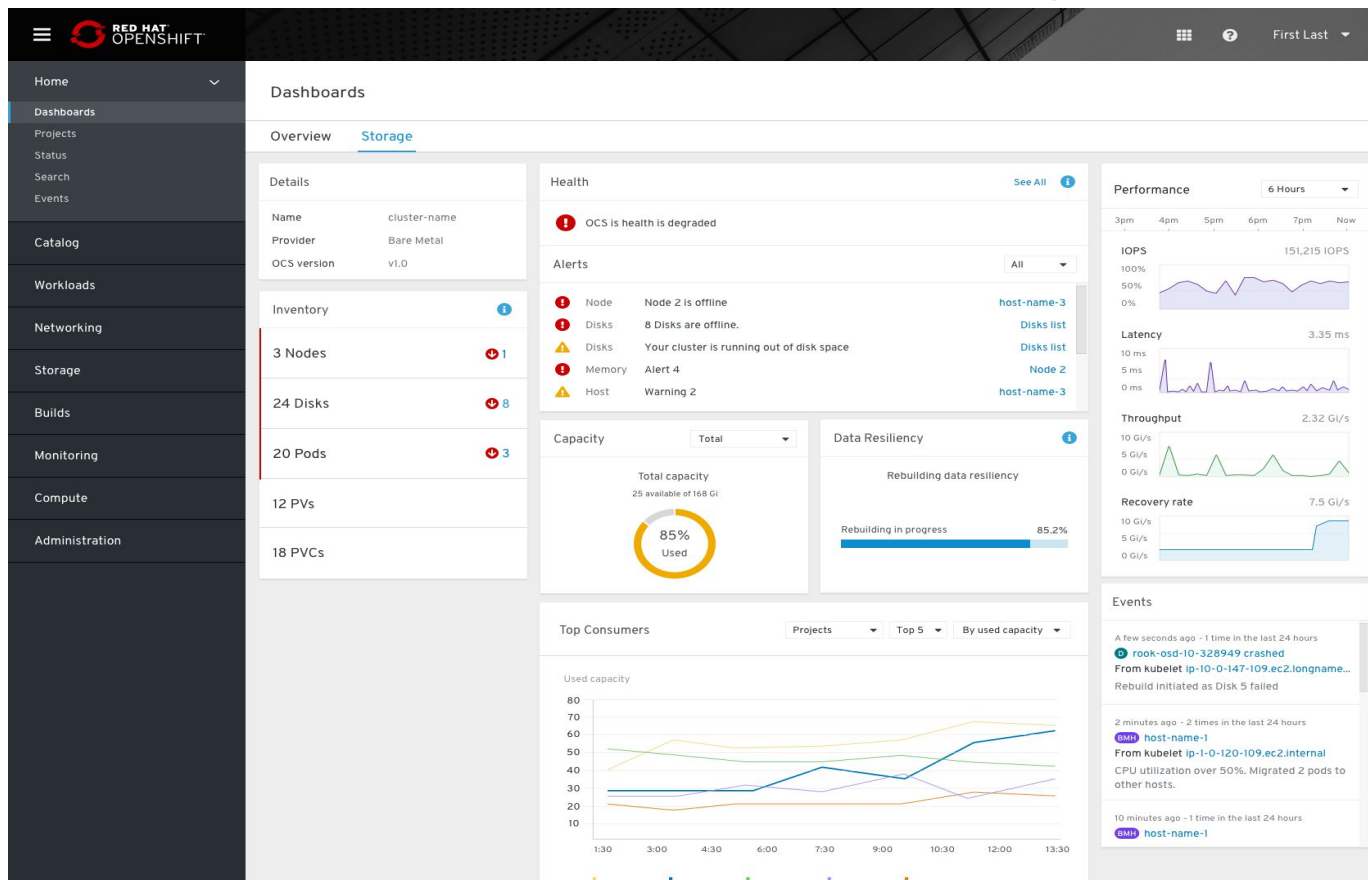
OCS for Persistent Storage

- New Storage backend for OCP 4 based on Ceph and Nooba
 - Adds enterprise S3 storage
- Install from Operator Hub
- Offers Ceph via Rook and Nooba as Multi-Cloud-Gateway
- Manages storage at scale
- Handles upgrades and health state
- Identifies availability zones and configures data replication accordingly
- Integrated with Openshift monitoring

OCS for Persistent Storage



OCS for Persistent Storage



App migration experience

Using open source tooling based on Velero

Velero is an upstream project previously known as Ark. Check out [this video](#) if you are curious and want to get a sneak peek at our capabilities.

What's moved during a migration

- Namespaces
- Persistent Volumes (move or copy)
- All important resource objects (Deployments, StatefulSets, etc)

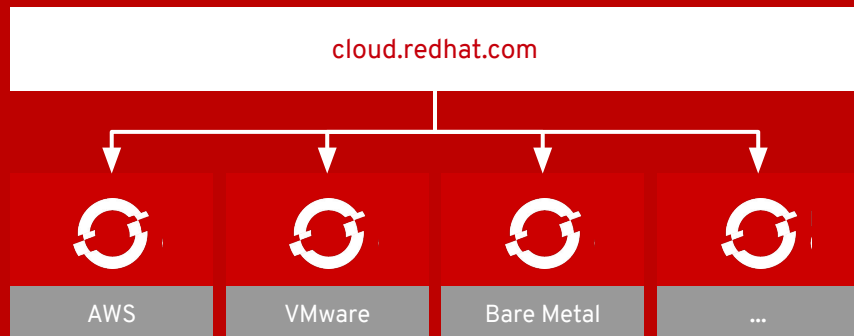
Available in OpenShift 4.2

The screenshot shows the 'Migration Plan Wizard' interface. The top section is titled 'Migration Plan Wizard' with the subtitle 'Create a migration plan'. Below this, there is a sidebar with five steps: 1 General, 2 Migration Source (selected), 3 Persistent Volumes, 4 Migration Targets, and 5 Results. The main content area shows the 'Source Cluster' dropdown set to 'Summit Demo Source Cluster'. Below this, there is a section 'Select projects to be migrated:' with a table listing projects. The table has two rows: 'robot-shop' (checked) and 'sandbox' (unchecked). Below the wizard, there is a table titled 'Migration Plans' showing a list of migration plans.

Name	Migrations	Source	Target	Repository	Persistent Volumes	Last Status
demo plan	2	Summit Demo Source Cluster	Target cluster	mydemobucket	2	Migrated Success
demo2	2	Summit Demo Source Cluster	Target cluster	mydemobucket	2	Migrated Success

Cloud-like Simplicity, Everywhere

Full-stack automated operations across any on-premises,
cloud, or hybrid infrastructure



CLOUD-LIKE SIMPLICITY, EVERYWHERE

OpenShift Cluster Manager cloud.redhat.com/openshift

Enhanced OpenShift Web Console Integration

Bi-directional navigation to and from the OpenShift web console for cluster administrators. Deep linking from OCM to the console where relevant.

OpenShift Dedicated cluster management

Self-service cluster provisioning, scaling, and basic management for OpenShift Dedicated customers (4.1+).

Cluster Monitoring

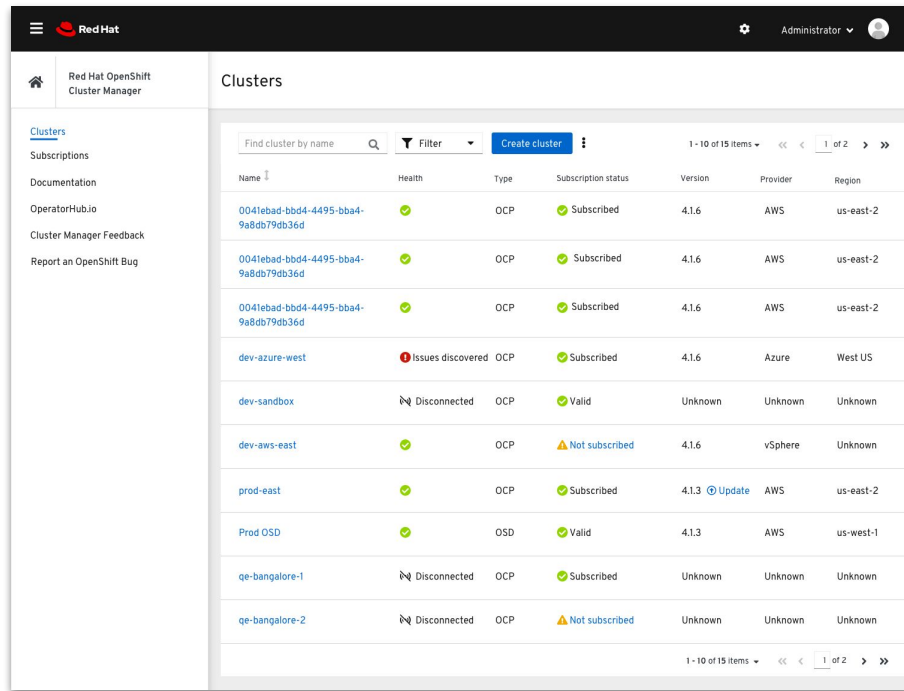
New tab available on all cluster detail pages helps cluster administrators discover critical issues impacting their clusters.

Cluster Updates & New Metrics

OCM provides a link to your cluster's settings page when updates are available for your cluster. Infrastructure provider and region are now captured and displayed for all clusters, where available.

More OpenShift Install Options

New infrastructure providers, including CodeReady Containers, are now listed as install options.



The screenshot shows the Red Hat OpenShift Cluster Manager interface. The left sidebar contains navigation links: Clusters (active), Subscriptions, Documentation, OperatorHub.io, Cluster Manager Feedback, and Report an OpenShift Bug. The main content area is titled 'Clusters' and features a search bar, a filter dropdown, a 'Create cluster' button, and pagination controls. Below these is a table listing various clusters with columns for Name, Health, Type, Subscription status, Version, Provider, and Region.

Name	Health	Type	Subscription status	Version	Provider	Region
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	Subscribed	4.1.6	AWS	us-east-2
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	Subscribed	4.1.6	AWS	us-east-2
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	Subscribed	4.1.6	AWS	us-east-2
dev-azure-west	Issues discovered	OCP	Subscribed	4.1.6	Azure	West US
dev-sandbox	Disconnected	OCP	Valid	Unknown	Unknown	Unknown
dev-aws-east	✓	OCP	Not subscribed	4.1.6	vSphere	Unknown
prod-east	✓	OCP	Subscribed	4.1.3	AWS	us-east-2
Prod OSD	✓	OSD	Valid	4.1.3	AWS	us-west-1
qe-bangalore-1	Disconnected	OCP	Subscribed	Unknown	Unknown	Unknown
qe-bangalore-2	Disconnected	OCP	Not subscribed	Unknown	Unknown	Unknown

Generally Available



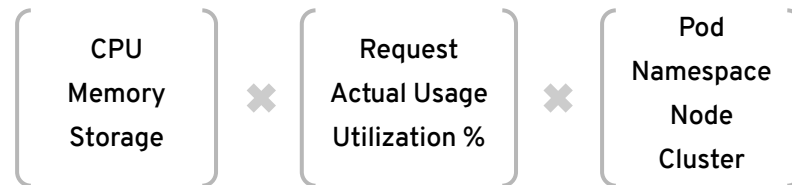
IMPROVE OBSERVABILITY

Metering

ShowBack/ChargeBack Reports available from OperatorHub

- Base functionality on all providers
- Tie into cloud providers for \$\$
- Included reports for 80% use-case
- Customers can write custom reports and time periods
- Offer basic UI reporting but main use is to plug into customer's BI tool of choice

Name	Namespace	Labels	Created At
cluster-cpu-capacity	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-capacity-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-usage	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-usage-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-utilization	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-capacity	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-capacity-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-usage	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-usage-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-utilization	openshift-metering	operator-metering=true	7 minutes ago



Generally Available

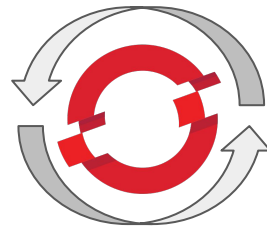
Cluster Logging

Overall performance improvements in collecting logs in 4.x w/ fluentd

- Tripled the amount of logs we collect and at the same time reduced resource consumption by ~50%

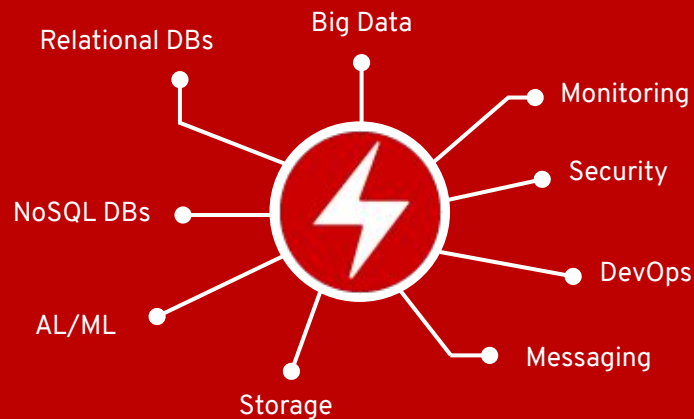
3.11	4.1
750 - 1000 1Kb messages/second/fluentd over 10 namespaces at ~600Mb RSS memory used and 75-85% utilization of a single core.	2250 - 2500 1Kb messages/second/fluentd over 10 namespaces at ~325Mb RSS memory used and 35% utilization of a single core.

Be notified when something is wrong with Fluentd with new alerting rules in OCP 4.2.



A broad ecosystem of workloads

Operator-backed services allow for a
SaaS experience on your own infrastructure

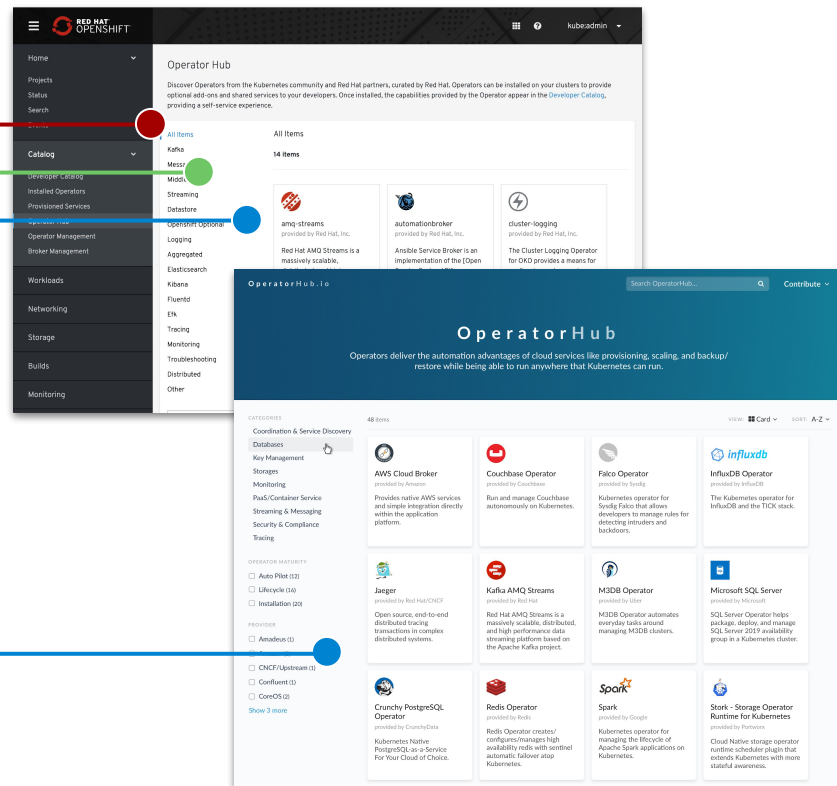


OPERATORHUB

- Accessible to admins only
- Discovery/install of all optional components and apps
- Upstream and downstream content
- ISV partners will support their Operators

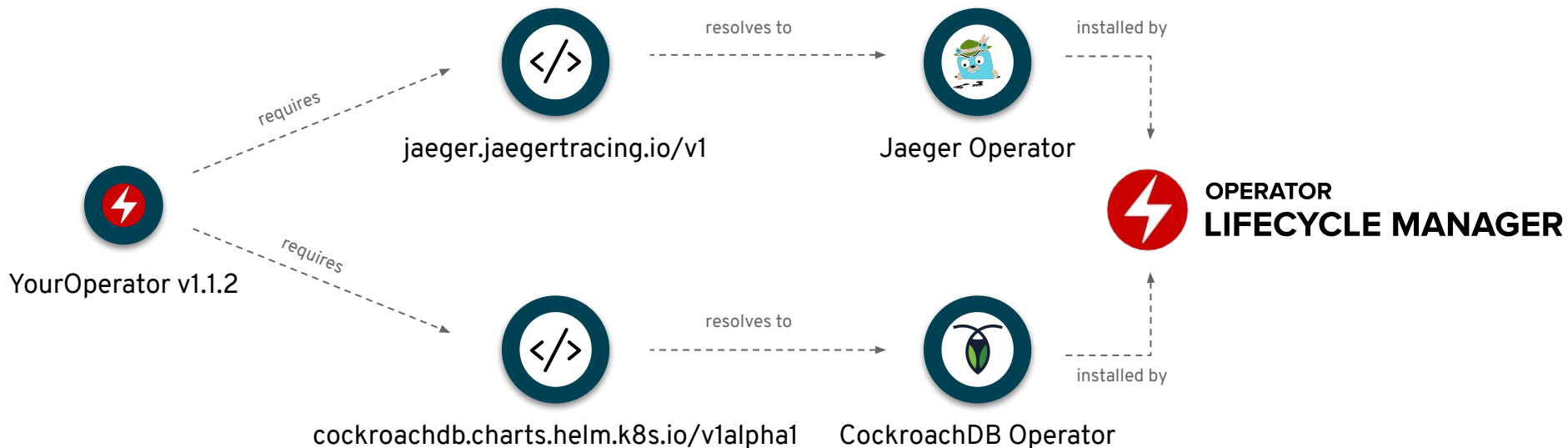
TYPES OF OPERATORS

Red Hat Products
ISV Partners
Community



4.2 Automated Dependency Resolution

Operator Framework Dependency Graphs



OpenShift Console

The future is now.

**Extending the
Console**

**Improve
Observability**

**Administration
made easy**

**Scaling your
Cluster**

Expose Third Party App Console for Operator-backed Services

Expose Operator-backed service Console through console CRD

Easily integrate/onboard third-party user interfaces in order to develop, administer, and configure Operator-backed services.

The screenshot displays the Red Hat OpenShift Container Platform console interface. The top navigation bar includes the Red Hat logo, the text "OpenShift Container Platform", and a user profile dropdown for "kubernetes:admin". A notification banner at the top states: "This is an example notification message with an optional link. [Optional link text](#)".

The left sidebar contains a menu with the following items: Administrator, Home, Operators (expanded), OperatorHub, Installed Operators (selected), Workloads, Serverless, Networking, Storage, Builds, Monitoring, and Compute.

The main content area shows the "Installed Operators" page for the "Project: tony" namespace. It includes a description: "Installed Operators are represented by Cluster Service Versions within this namespace. For more information, see the [Operator Lifecycle Manager documentation](#). Or create an Operator and Cluster Service Version using the [Operator Lifecycle Manager](#)." Below this is a table of installed operators.

A dropdown menu is open, showing options for "Red Hat Applications" (including OpenShift Cluster Manager) and "Third Party Applications" (including Couchbase Server Web Console). A blue dotted line connects the "Couchbase Server Web Console" option in the dropdown to the "Couchbase Operator" entry in the table.

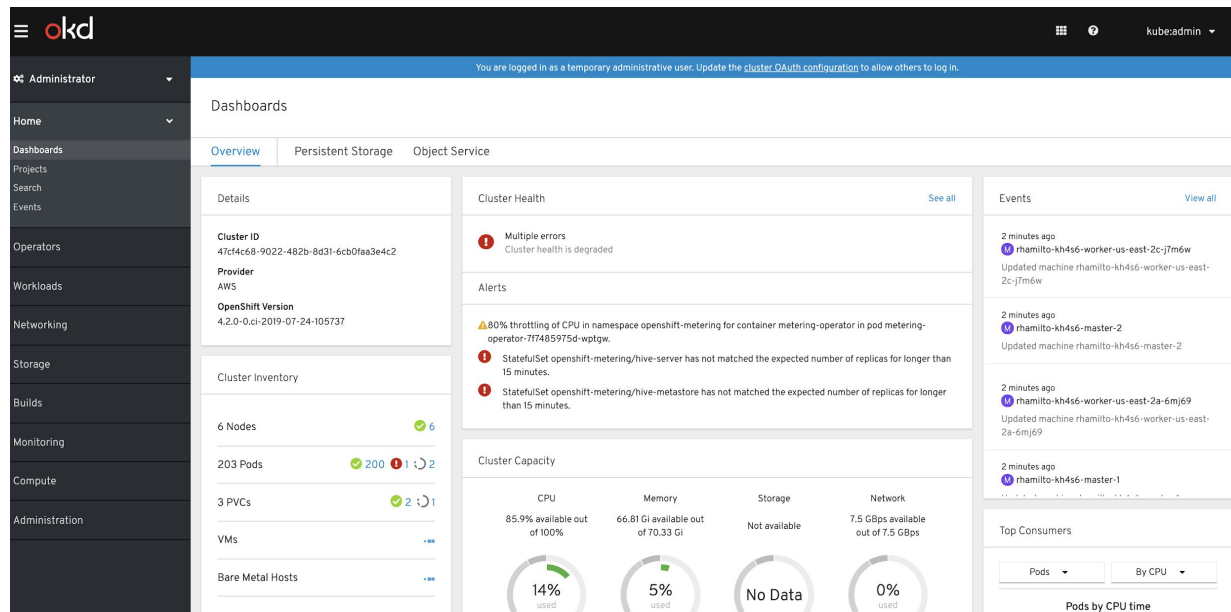
Name	Namespace	Deployment	Status	Provided APIs
AMQ Streams 1.1.0 provided by Red Hat, Inc.	NS tony	amq-streams-cluster-operator	InstallSucceeded Up to date	Kafka Kafka Connect Kafka Connect S2I Kafka MirrorMaker View 2 more...
Couchbase Operator 1.1.0 provided by Couchbase	NS tony	couchbase-operator	InstallSucceeded Up to date	Couchbase Cluster
etcd 0.9.4 provided by CNCF	NS tony	etcd-operator	InstallSucceeded Up to date	etcd Cluster etcd Backup etcd Restore
Knative Serving Operator 0.7.1 provided by Red Hat	NS tony	knative-serving-operator	InstallSucceeded Up to date	Knative Serving

Enhanced Visibility with the New Dashboard

Cluster-wide Dashboard gives Admins Clear Insights

Drill down in context from the new dashboard widgets:

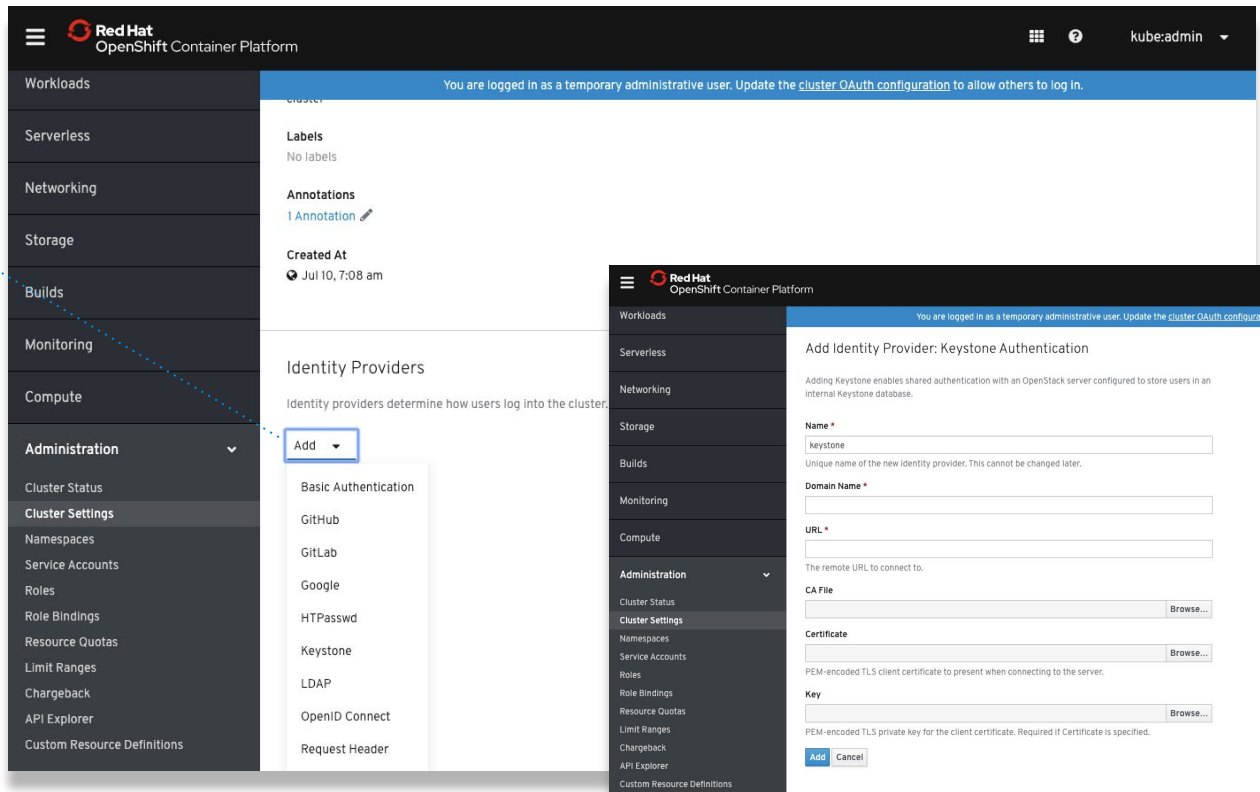
- Cluster Details
- Cluster Health
- Cluster Inventory
- Cluster Capacity
- Cluster Utilization
- Top Consumers



Configuring Authentication for your desired Identity Providers

Customize and determine how users log into the cluster

- Basic Authentication
- GitHub
- GitLab
- Google
- HTPasswd
- Keystone
- LDAP
- OpenID Connect
- Request Header

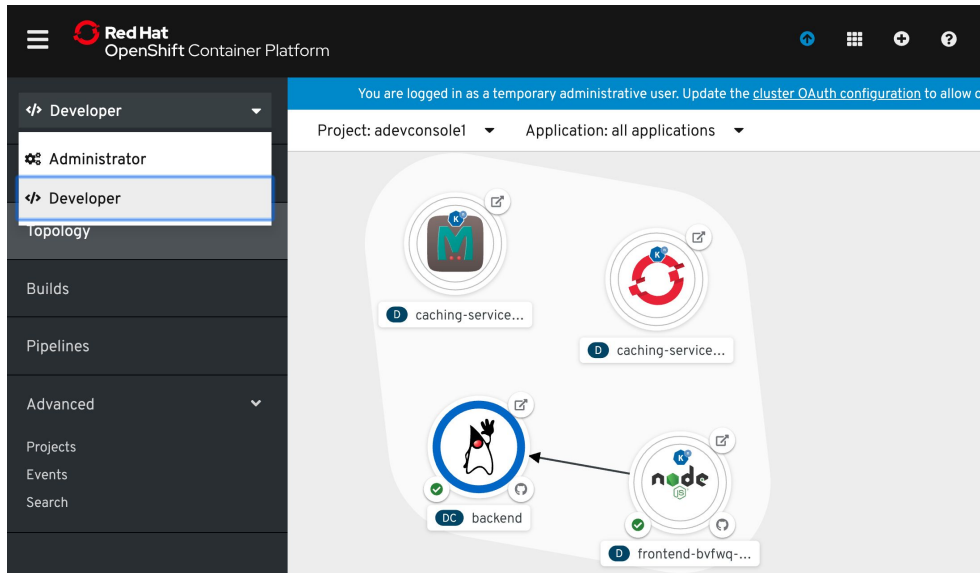


Web console - Developer Perspective

Key Features

An alternative perspective in the OpenShift UI that will sit beside the admin console and focus on developer use cases.

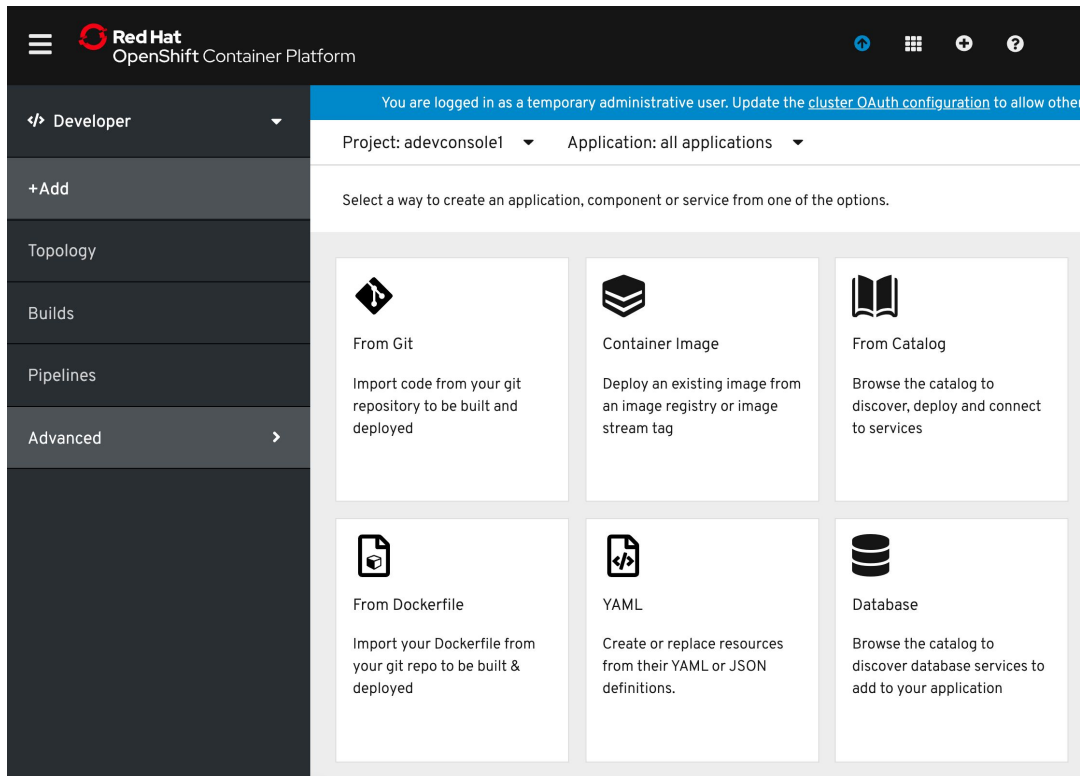
All OpenShift developer tool UIs will be surfaced here...though some (like CodeReady Workspaces) will be links out to unique UIs.



Easily Create an Application

Key Features

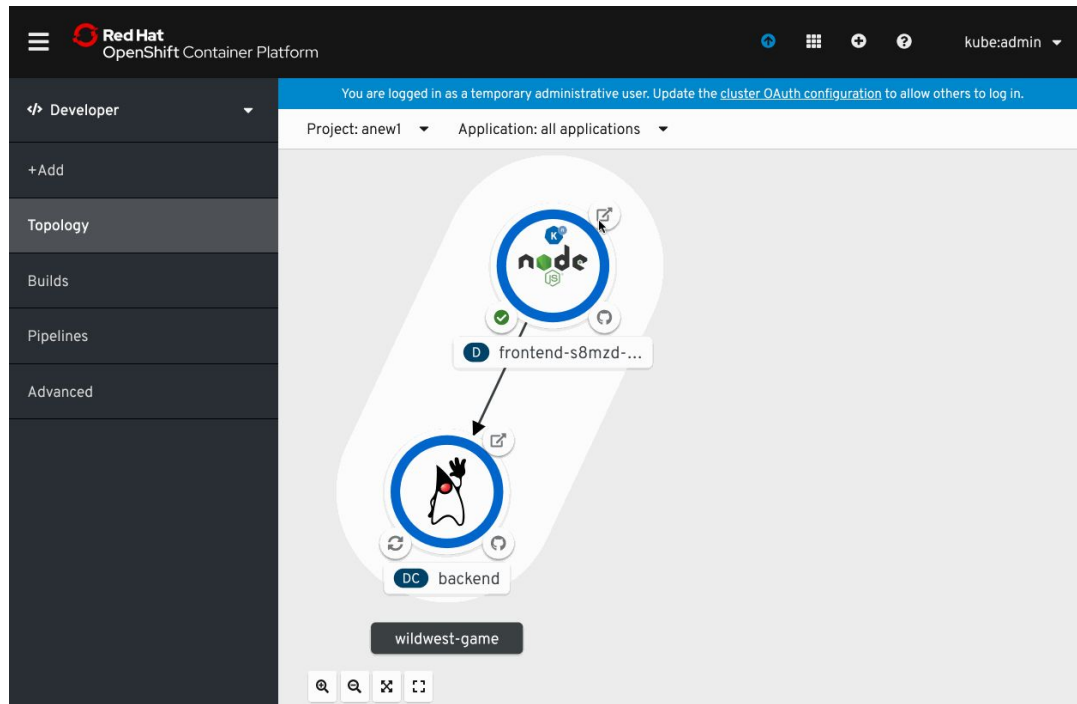
- Source in Git
- Existing container image
- YAML definition
- Build from Dockerfile
- Explore catalog
- Databases from catalog



Application Topology

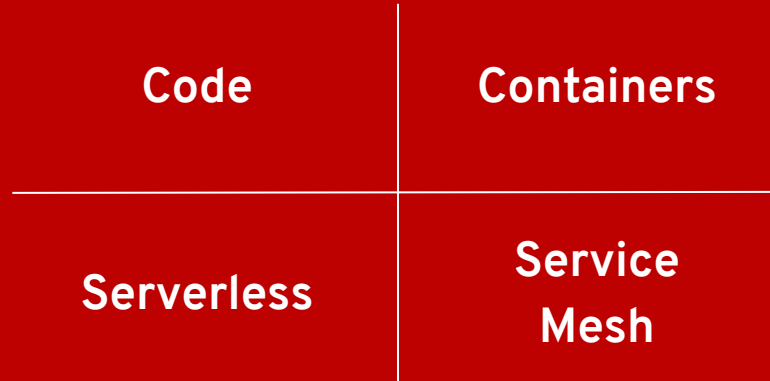
Key Features

- View structure and status of app components
- Drill into specific workloads
- Quickly navigate to pod logs
- Manually scale
- Pod donut!
- Access route/URL
- Linked build and source



Next wave of developer tools

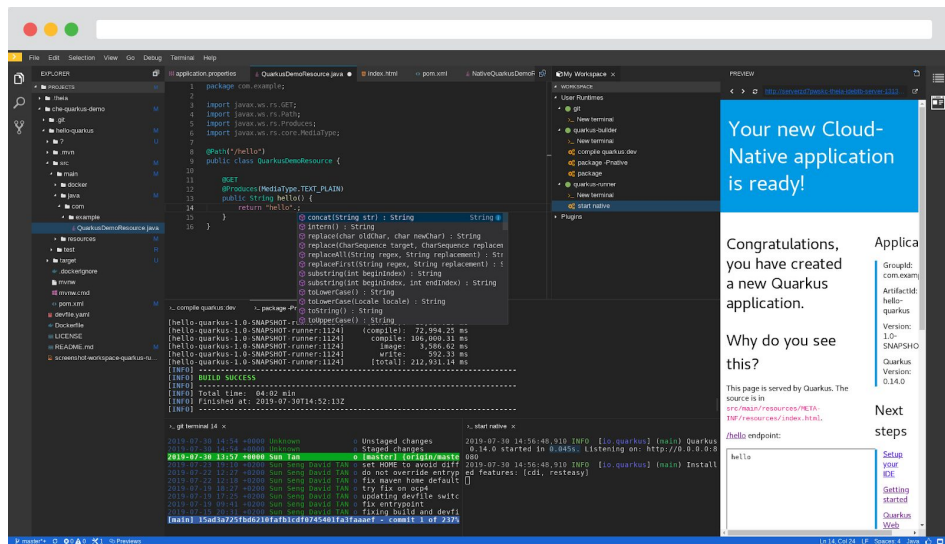
OpenShift has all of the latest tools to make
your devs more productive



CodeReady Workspaces 2.0

Based on Eclipse Che 7

- **Kubernetes-based developer workspaces:** Fully containerized developer workspaces allowing to bring your K8S application runtime easily in your dev environment.
- **New Editor:** New default web-based editor provides a VSCode like experience in the browser.
- **Devfile:** Configure a devfile for your project and get reproducible and portable developer environments.
- **VSCode plug-ins compatibility**
- **Swappable Editor**
- **OpenShift VSCode Plug-in**
- **Easier to Monitor and Administrate:** Prometheus and Grafana dashboards.



Shipped independently from 4.2, shortly after

CodeReady Containers: OpenShift on your Laptop

Provides a pre-built development environment based on **Red Hat Enterprise Linux** and **OpenShift** for quick container-based application development. Use with OpenShift on-premises or cloud.

```
$ crc setup
Prepare your machine for running OpenShift

$ crc start -b
crc-hyperkit-4.2.0.crcbundle
Start with the Hyperkit 4.2 bundle

$ crc status
Get the status of the cluster
```

- Based on OpenShift 4.x
- Linux (libvirt)
- Windows (Hyper-V)
- MacOS (Virtualbox)
- External beta available
- Replaces the 3.x experiences around:
 - Minishift
 - CDK
 - oc cluster up

Use It To: Simplify direct-to-OpenShift development on laptops.

Cloud-native CI/CD with OpenShift Pipelines

- Based on Tekton Pipelines
- Runs serverless (no babysitting!)
- Containers as building blocks
- Deploy to multiple platforms
- Standard CRDs
- Build images with Kubernetes tools (s2i, buildah, kaniko, jib, buildpack, etc)
- Pipelines portable to any Kubernetes
- Available in OperatorHub

The screenshot displays the Red Hat OpenShift Container Platform interface for managing Pipelines. The left sidebar shows navigation options: Developer, Add, Topology, Builds, Pipelines (selected), and Advanced. The main content area shows the 'Project: Project01' and 'Pipeline Run Details' for 'pipelinerun01a', which is in a 'Running' state. A 'Tech Preview' badge is visible. The 'Overview' tab is active, showing a 'Pipeline Run Overview' diagram with steps: code compile, compile & test, unit test, security check, and image build. Below the diagram, details for the pipeline run are listed: Name (pipelinerun01a), Namespace (project01), Labels (app=dummy-mongo-pod-test, bap.me/environment=dev, bap.me/track=experimental, bap.me/tier=backend), Annotations, and Created At. The 'Logs' tab is also visible, showing a log entry for the 'image build' step, indicating a successful build process.



OpenShift Serverless

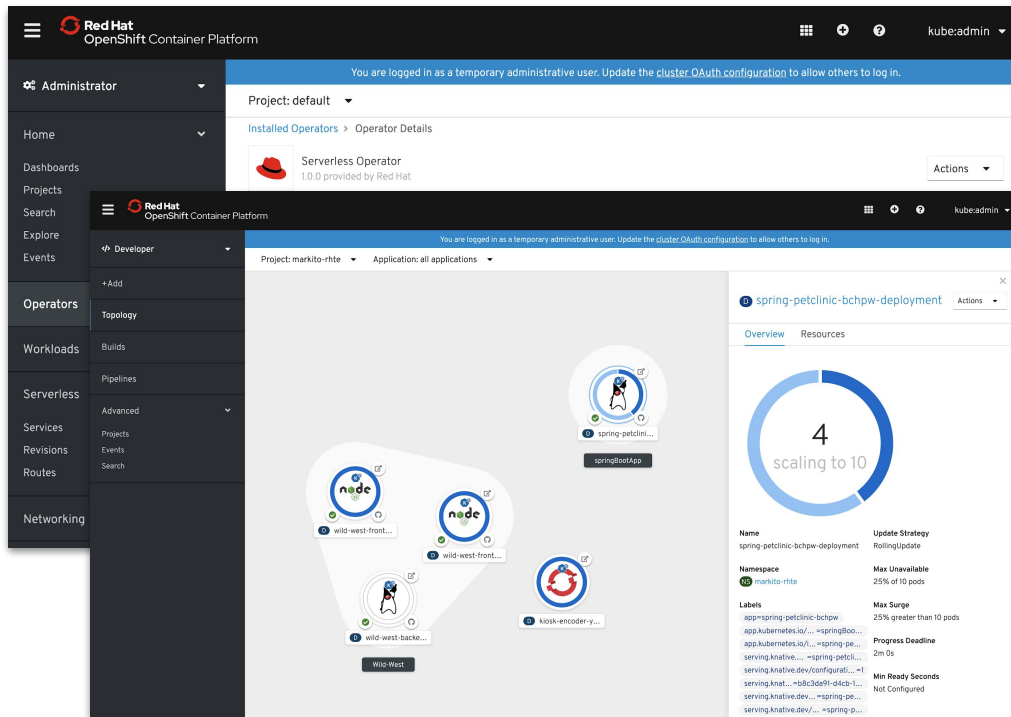
Key Features

- Familiar to Kubernetes users. Native.
- Scale to 0 and autoscale to N based on demand
- Applications and functions. Any container workload.
- Powerful eventing model with multiple event sources.
- Operator available via OperatorHub
- Knative v0.7.1 (v1beta1 APIs)
- No vendor lock in

Learn more

<https://openshift.com/learn/topics/serverless>

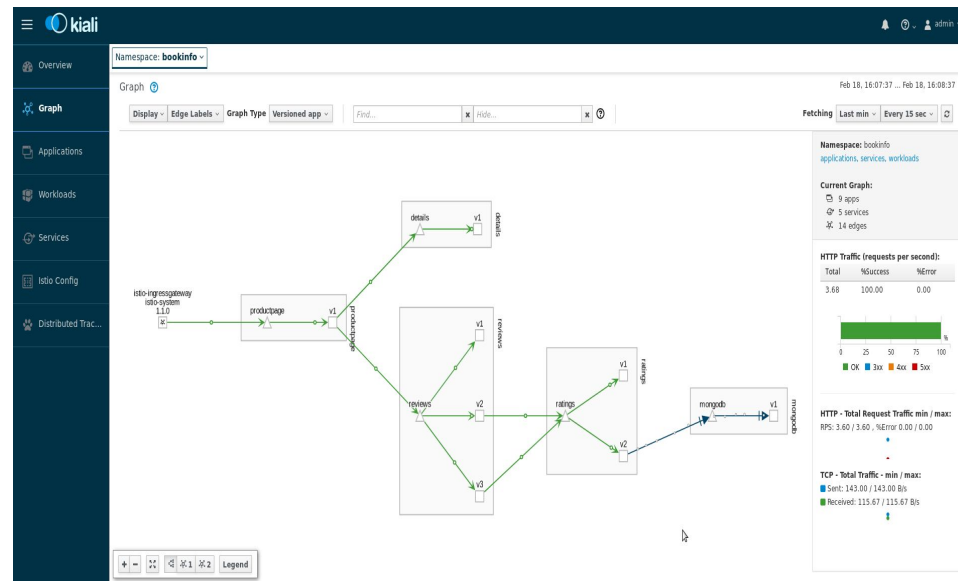
<https://redhat-developer-demos.github.io/knative-tutorial>



OpenShift Service Mesh

Key Features

- A dedicated network for service to service communications
- Observability and distributed tracing
- Policy-driven security
- Routing rules & chaos engineering
- Powerful visualization & monitoring
- Available via OperatorHub



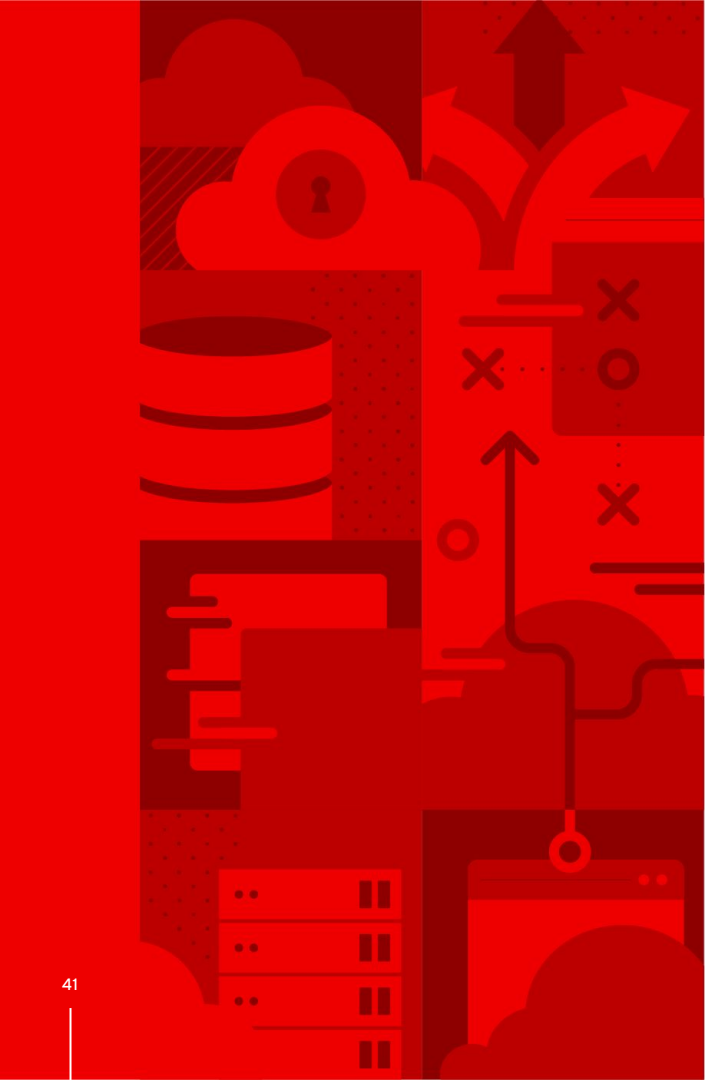
Roadmap

2019 Roadmap

Q2 CY2019 OpenShift 4.1	
DEV	- OpenShift Serverless (Knative) - DP - OpenShift Pipelines (Tekton) DP2 - CodeReady Workspaces - CodeReady Containers Alpha - Developer CLI (odo) Beta
APP	- OperatorHub - Operator Lifecycle Manager - Service Mesh (~4 month after)
PLATFORM	- Kubernetes 1.13 with CRI-O runtime - RHEL CoreOS, RHEL7 - Automated Installer for AWS - Pre-existing Infra Installer for Bare Metal, VMware, AWS - Automated, one-click updates - Multus (Kubernetes multi-network) - Quay v3
HOSTED	- cloud.redhat.com - Multi-Cluster Mgmt - OCP Cluster Subscription Management - Azure Red Hat OpenShift - OpenShift Dedicated consumption pricing

Q3 CY2019 OpenShift 4.2	
DEV	- Developer Console GA - OpenShift Serverless (Knative) - TP - OpenShift Pipelines (Tekton) DP3 - CodeReady Containers GA - Developer CLI (odo) GA
APP	- OperatorHub Enhancements - Operator Deployment Field Forms - Application Migration Console
PLATFORM	- Kubernetes 1.14 w/ CRI-O runtime - Disconnected Install and Update - Automated Installer for Azure, OSP, GCP - Pre-existing Infra Installer for GCP - Cluster-wide Egress Proxy - OVN Tech Preview - OpenShift Container Storage 4.2 (1 month after)
HOSTED	Insights Operator

Q4 CY19/Q1 CY20 OpenShift 4.3	
DEV	- OpenShift Serverless (Knative) - GA - OpenShift Pipelines (Tekton) TP - Helm 3 TP
APP	- Metering for Services - Windows Containers (Planned) - GPU Metering - Application Operator Binding - DP
PLATFORM	- Kubernetes 1.16 w/ CRI-O runtime - Automated Installer for RHV - Private/Internal Clusters support from the installer - Deploy to pre-existing VPC & Subnets - OVN GA w/ Windows Networking Integration (Planned) - FIPS - Pre-existing Infra Installer for OSP - OpenShift Container Storage 4.3
HOSTED	cloud.redhat.com - Subscription Mgmt Improvements

A vertical red graphic on the left side of the slide. It contains various white icons: a cloud with a keyhole, a database cylinder, a server rack, a computer monitor, and several arrows pointing in different directions, some with 'X' marks. The background of the entire slide is a light gray.

Questions?

try.openshift.com



linkedin.com/company/red-hat



facebook.com/redhatinc



youtube.com/user/RedHatVideos



twitter.com/RedHat